ASSIGNMENT 3

part 1: Controller Hub and Learning Switch

In this scenario, the latencies of the three ping packets with the Hub controller appear nearly identical. However, the reason for the observed latency differences between a Hub Controller and a Learning Switch stems from their unique functionalities within network architecture.

The Hub Controller, as part of Software-Defined Networking (SDN), contributes to additional latency due to its centralized control and the management of policies. This results in increased processing and communication overhead as it operates to orchestrate network traffic.

On the other hand, the Learning Switch, typical in traditional Ethernet networks, excels in minimizing latency by leveraging its capability to learn from previous packets. It efficiently forwards traffic based on its dynamically updated MAC address table, thereby reducing delays in the transmission process.

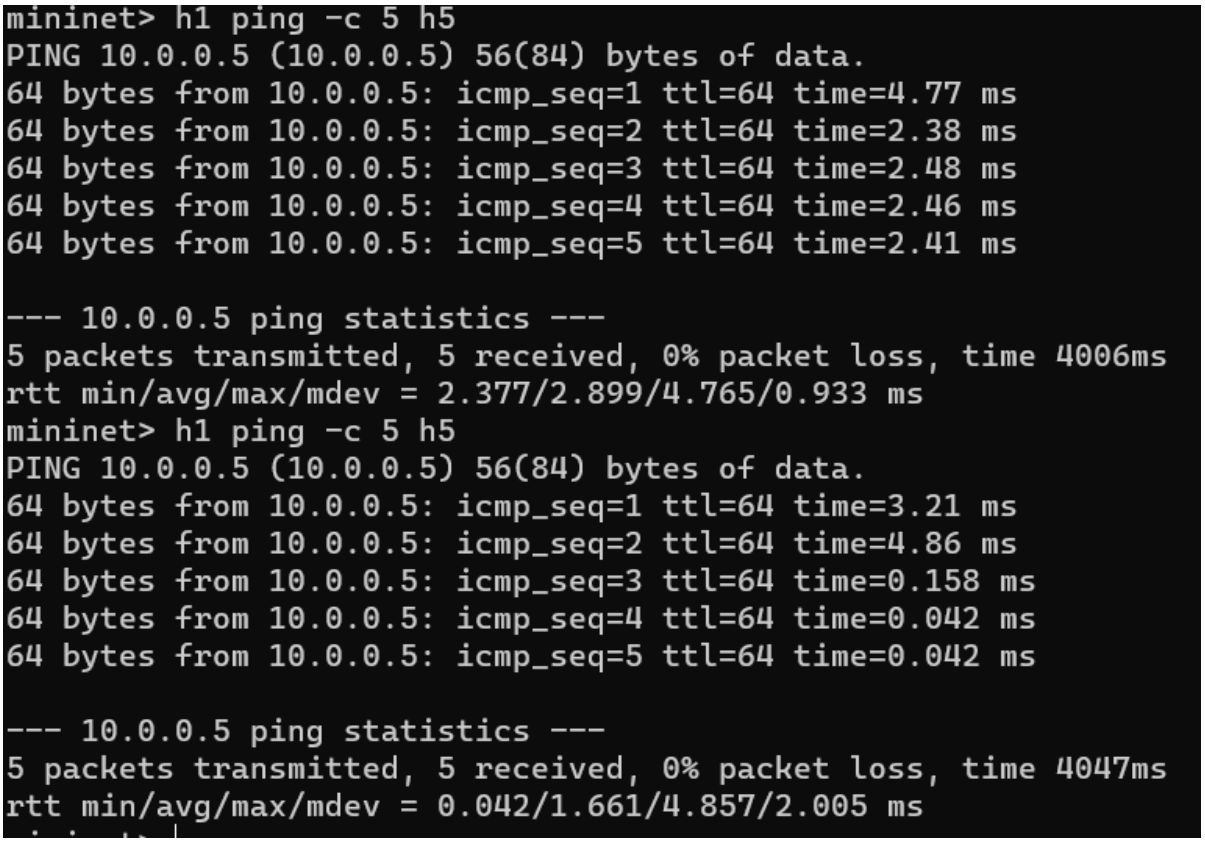
```
mininet> h2 ping -c 3 h5
PING 10.0.0.5 (10.0.0.5) 56(84) bytes of data.
64 bytes from 10.0.0.5: icmp_seq=1 ttl=64 time=2.48 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=64 time=2.31 ms
64 bytes from 10.0.0.5: icmp_seq=3 ttl=64 time=2.47 ms

--- 10.0.0.5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 2.309/2.417/2.476/0.076 ms
```

```
mininet@mininet-vm:~/A3$ ryu-manager learning_switch.py
loading app learning_switch.py
```

```
mininet@mininet-vm:~/A3$ ryu-manager controller_hub.py
```

```
mininet> h2 ping -c 3 h5
PING 10.0.0.5 (10.0.0.5) 56(84) bytes of data.
64 bytes from 10.0.0.5: icmp_seq=1 ttl=64 time=5.75 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=64 time=2.22 ms
64 bytes from 10.0.0.5: icmp_seq=3 ttl=64 time=0.200 ms
```



Throughput

The Learning Switch, commonly employed in Software-Defined Networking (SDN), optimizes throughput by utilizing a MAC address table to smartly route packets according to their destination addresses, effectively diminishing unnecessary broadcast transmissions. This method proves highly efficient in maximizing throughput as it selectively directs packets solely to their intended destinations, which is especially advantageous in larger, less congested networks. Nonetheless, the process demands additional processing overhead to manage the MAC address table and might encounter constraints when dealing with exceptionally high traffic volumes. Therefore, the decision between employing the Learning Switch and the alternative methods relies on various factors such as the size of the network, patterns of traffic, and specific performance necessities.

```
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 h4 h5
h2 -> h1 h3 h4 h5
h3 -> h1 h2 h4 h5
h4 -> h1 h2 h3 h5
h5 -> h1 h2 h3 h4
*** Results: 0% dropped (20/20 received)
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 h4 h5
h2 -> h1 h3 h4 h5
h3 -> h1 h2 h4 h5
h4 -> h1 h2 h3 h5
h5 -> h1 h2 h3 h4
*** Results: 0% dropped (20/20 received)
mininet>
```

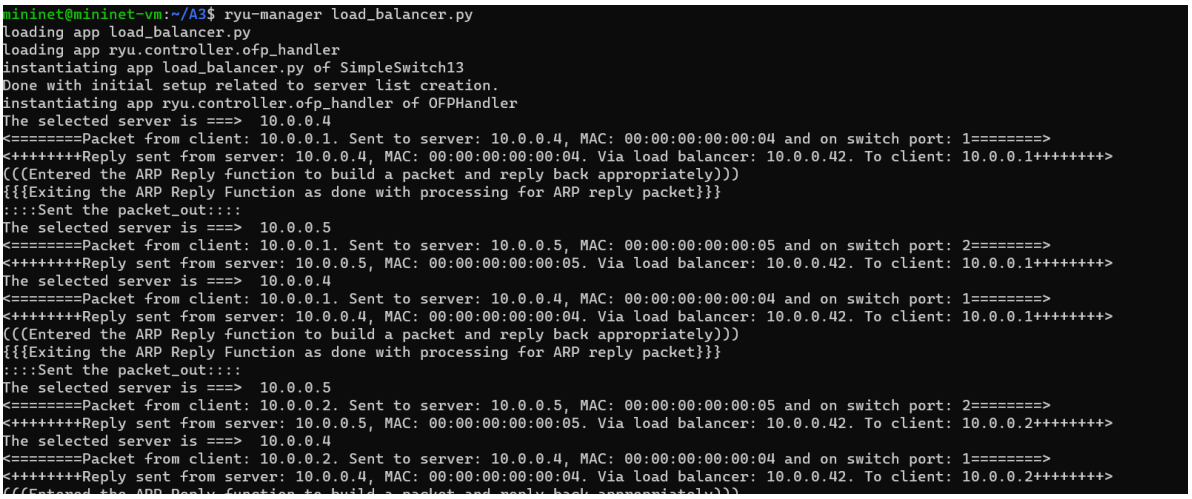
part 2: Firewall and Monitor

```
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 X h5
h2 -> h1 h3 h4 X
h3 -> h1 h2 h4 X
h4 -> X h2 h3 h5
h5 -> h1 X X h4
*** Results: 30% dropped (14/20 received)
```

The network operator's objective is to restrict communication between specific hosts—H2 and H3 with H5, and H1 with H4—by deploying a rule that drops packets. This rule functions by identifying packets with a source IP address or MAC address corresponding to one host and a destination IP address or MAC address matching another host, effectively preventing their communication.

To ensure the effectiveness of the real-time firewall policy and prevent interference from pre-existing rules, several measures should be adopted. First, assigning higher priority levels to firewall rules is crucial. Additionally, establishing a clear and well-defined rule evaluation order is essential. Implementing conflict detection and resolution mechanisms further enhances the efficiency of the firewall. Simultaneously, continuous monitoring and thorough auditing of rule changes play a critical role in maintaining consistency within the network's security measures.

part 3: Load Balancer



For implementing a load balancing policy that accounts for server loads, in addition to the current round-robin distribution when pinging 10.0.0.42, it's essential to introduce a system that monitors server performance at regular intervals. This dynamic monitoring allows for the distribution of incoming traffic to servers with lower loads, thus ensuring more efficient resource utilization. This proactive approach not only optimizes server usage but also enhances the overall user experience by directing requests to servers that are better positioned to handle them at any given time.